Contour Plots & Gradients

Nipun Batra and teaching staff

IIT Gandhinagar

August 17, 2025

Table of Contents

Understanding Contour Plots

Introduction to Contour Plots

Definition: What is a Contour Plot?

Concept: A contour plot shows curves where a function $f(x, y) = K$ for different constant values K

Introduction to Contour Plots

Definition: What is a Contour Plot?

Concept: A contour plot shows curves where a function $f(x, y) = K$ for different constant values K

Example: Function: $z = f(x, y) = x^2 + y^2$

Circular Contours

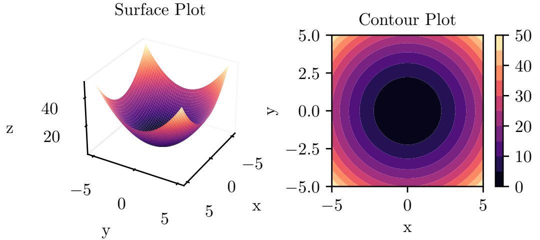
Introduction to Contour Plots

Definition: What is a Contour Plot?

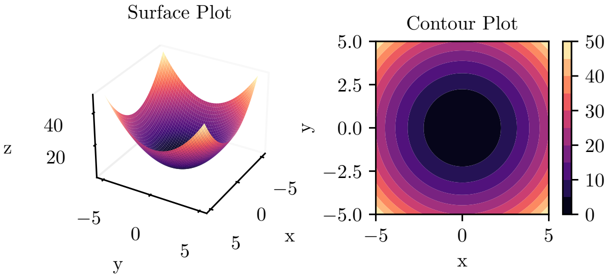
Concept: A contour plot shows curves where a function $f(x, y) = K$ for different constant values K

Example: Function: $z = f(x, y) = x^2 + y^2$

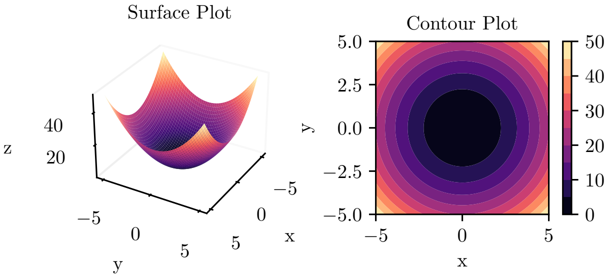
Circular Contours



Introduction to Contour Plots



Introduction to Contour Plots



Key Points

Key Insight: Each contour line represents all points (x, y) where $f(x, y) = K$ for a specific constant K

Contour Example: Parabolic Function

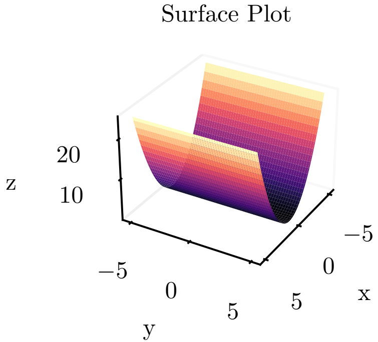
Example: Function: $z = f(x, y) = x^2$

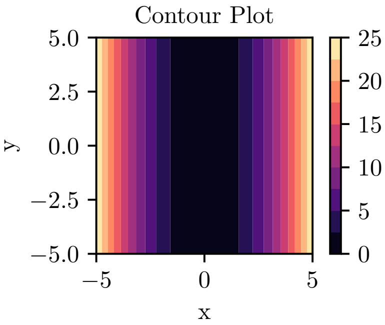
Note: This function depends only on x, not on y!

Contour Example: Parabolic Function

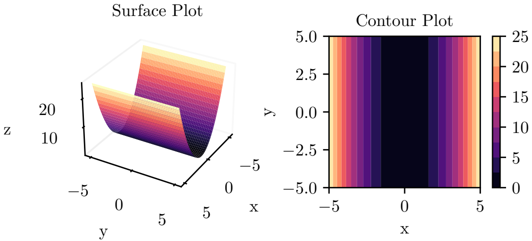
Example: Function: $z = f(x, y) = x^2$

Note: This function depends only on x, not on y!





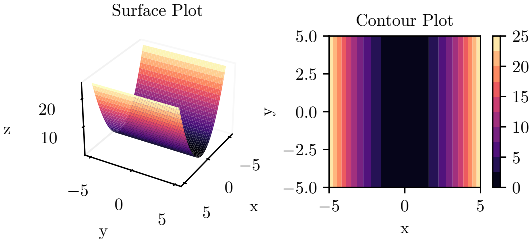
Contour Example: Parabolic Function



Key Points

Observation: Contour lines are vertical because $f(x, y) = x^2$ is constant for all y values when x is fixed

Contour Example: Parabolic Function



Key Points

Observation: Contour lines are vertical because $f(x, y) = x^2$ is constant for all y values when x is fixed

Important: ML Connection

This represents: A loss function that doesn't depend on one of the parameters!

Contour Example: Manhattan Distance

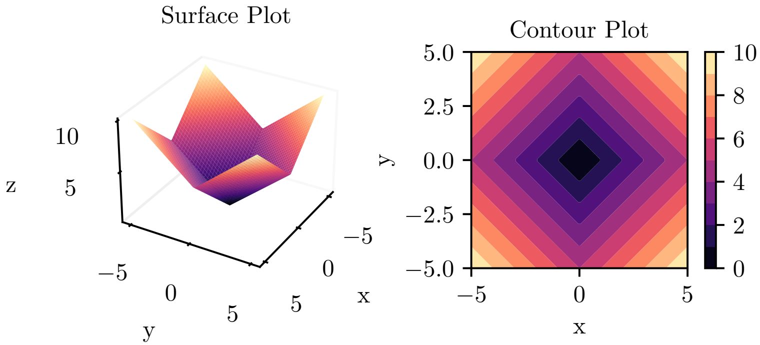
Example: Function: $z = f(x, y) = |x| + |y|$

Also known as: Manhattan distance or L1 norm

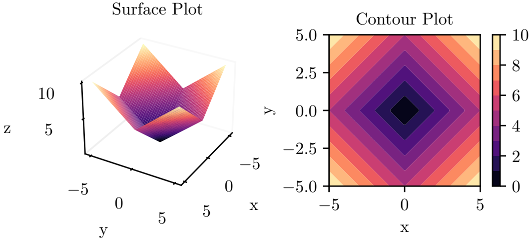
Contour Example: Manhattan Distance

Example: Function: $z = f(x, y) = |x| + |y|$

Also known as: Manhattan distance or L1 norm



Contour Example: Manhattan Distance



Key Points

Shape: Diamond-shaped contours due to absolute value functions

Important: ML Connection

This represents: L1 regularization in machine learning (promotes sparsity!)

Contour Example: Polynomial Function

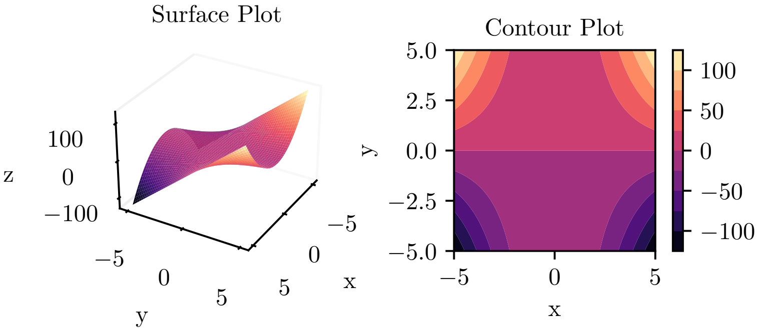
Example: Function: $z = f(x, y) = x^2 \cdot y$

Type: Mixed polynomial (quadratic in x, linear in y)

Contour Example: Polynomial Function

Example: Function: $z = f(x, y) = x^2 \cdot y$

Type: Mixed polynomial (quadratic in x, linear in y)



Contour Example: Polynomial Function

Key Points

Key Features:

- Asymmetric contours

Contour Example: Polynomial Function

Key Points

Key Features:

- Asymmetric contours
- Different behavior above and below $y = 0$

Contour Example: Polynomial Function

Key Points

Key Features:

- Asymmetric contours
- Different behavior above and below $y = 0$
- Non-linear interaction between variables

Contour Example: Polynomial Function

Key Points

Key Features:

- Asymmetric contours
- Different behavior above and below $y = 0$
- Non-linear interaction between variables

Important: ML Connection

This represents: Complex loss surfaces with variable interactions

Contour Example: Hyperbolic Function

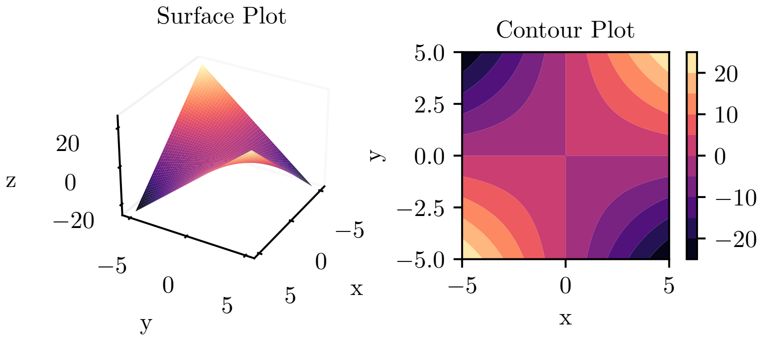
Example: Function: $z = f(x, y) = xy$

Type: Bilinear function (linear in each variable separately)

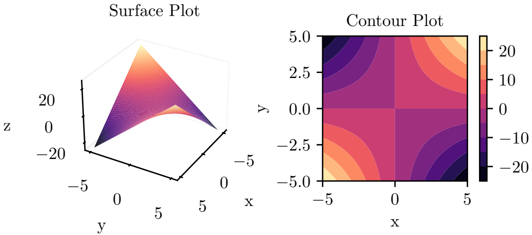
Contour Example: Hyperbolic Function

Example: Function: $z = f(x, y) = xy$

Type: Bilinear function (linear in each variable separately)



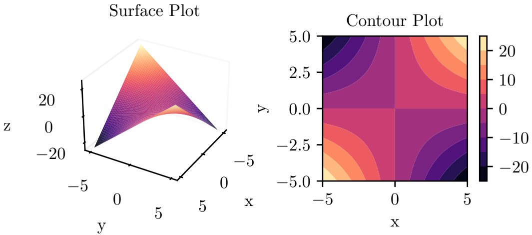
Contour Example: Hyperbolic Function



Key Points

Shape: Hyperbolic contours with saddle point at the origin

Contour Example: Hyperbolic Function



Key Points

Shape: Hyperbolic contours with saddle point at the origin

Important: ML Significance

Saddle points: Common in neural network optimization - neither minimum nor maximum!

Gradients and Contour Plots

Understanding Gradients

Definition: What is a Gradient?

Gradient ∇f: Vector pointing in the direction of steepest increase of function f

Understanding Gradients

Definition: What is a Gradient?

Gradient ∇f: Vector pointing in the direction of steepest increase of function f

Key Points

Key Properties

- **Direction:** Points toward steepest ascent

Understanding Gradients

Definition: What is a Gradient?

Gradient ∇f: Vector pointing in the direction of steepest increase of function f

Key Points

Key Properties

- **Direction:** Points toward steepest ascent
- **Magnitude:** Rate of steepest change

Understanding Gradients

Definition: What is a Gradient?

Gradient ∇f: Vector pointing in the direction of steepest increase of function f

Key Points

Key Properties

- **Direction:** Points toward steepest ascent
- **Magnitude:** Rate of steepest change
- **Contour relationship:** Always perpendicular to contour lines

Understanding Gradients

Example: Fundamental Insight

All points on the same contour have identical $f(x, y)$ values

Understanding Gradients

Example: Fundamental Insight

All points on the same contour have identical $f(x, y)$ values
Moving along a contour: No change in function value

Understanding Gradients

Example: Fundamental Insight

All points on the same contour have identical $f(x, y)$ values
Moving along a contour: No change in function value
Moving perpendicular to contour: Maximum change in function value

Understanding Gradients

Example: Fundamental Insight

All points on the same contour have identical $f(x, y)$ values
Moving along a contour: No change in function value
Moving perpendicular to contour: Maximum change in function value

Important: ML Application

Gradient descent: Move opposite to gradient direction to minimize loss!

Gradients Visualized: Circular Contours

Example: Function: $z = f(x, y) = x^2 + y^2$

Gradients Visualized: Circular Contours

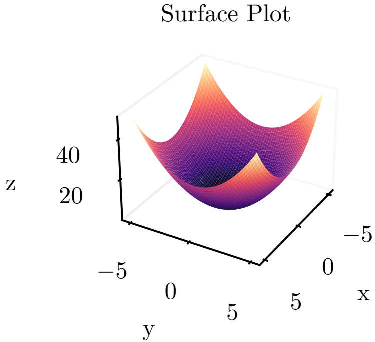
Example: Function: $z = f(x, y) = x^2 + y^2$

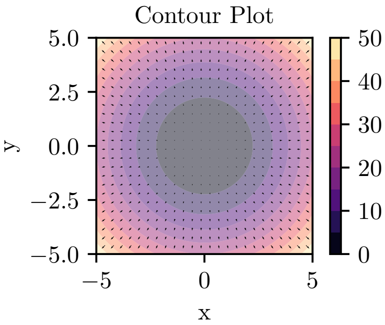
Gradient: $\nabla f = \begin{bmatrix} 2x \\ 2y \end{bmatrix}$

Gradients Visualized: Circular Contours

Example: Function: $z = f(x, y) = x^2 + y^2$

Gradient: $\nabla f = \begin{bmatrix} 2x \\ 2y \end{bmatrix}$





Gradients Visualized: Circular Contours

Key Points

Observations:

- Gradient arrows point radially outward

Gradients Visualized: Circular Contours

Key Points

Observations:

- Gradient arrows point radially outward
- Arrows are perpendicular to circular contours

Gradients Visualized: Circular Contours

Key Points

Observations:

- Gradient arrows point radially outward
- Arrows are perpendicular to circular contours
- Magnitude increases away from origin

Gradients Visualized: Circular Contours

Key Points

Observations:

- Gradient arrows point radially outward
- Arrows are perpendicular to circular contours
- Magnitude increases away from origin
- All arrows point toward steepest ascent

Gradients Visualized: Circular Contours

Key Points

Observations:

- Gradient arrows point radially outward
- Arrows are perpendicular to circular contours
- Magnitude increases away from origin
- All arrows point toward steepest ascent

Important: Perfect for Optimization

This is an ideal optimization landscape: Single global minimum at origin!

Gradient Properties: Key Insights

Important: Direction Interpretation

Gradient Properties: Key Insights

Important: Direction Interpretation

Steepest Ascent: Gradient ∇f points toward maximum increase in $f(x, y)$

Gradient Properties: Key Insights

Important: Direction Interpretation

Steepest Ascent: Gradient ∇f points toward maximum increase in $f(x, y)$
Steepest Descent: $-\nabla f$ points toward maximum decrease in $f(x, y)$

Gradient Properties: Key Insights

Important: Direction Interpretation

Steepest Ascent: Gradient ∇f points toward maximum increase in $f(x, y)$
Steepest Descent: $-\nabla f$ points toward maximum decrease in $f(x, y)$

Key Points

Contour Relationship

- **Same contour:** All points have identical $f(x, y)$ values

Gradient Properties: Key Insights

Important: Direction Interpretation

Steepest Ascent: Gradient ∇f points toward maximum increase in $f(x, y)$
Steepest Descent: $-\nabla f$ points toward maximum decrease in $f(x, y)$

Key Points

Contour Relationship

- **Same contour:** All points have identical $f(x, y)$ values
- **Gradient direction:** Always perpendicular to contour lines

Gradient Properties: Key Insights

Important: Direction Interpretation

Steepest Ascent: Gradient ∇f points toward maximum increase in $f(x, y)$
Steepest Descent: $-\nabla f$ points toward maximum decrease in $f(x, y)$

Key Points

Contour Relationship

- **Same contour:** All points have identical $f(x, y)$ values
- **Gradient direction:** Always perpendicular to contour lines
- **Zero gradient:** Occurs at critical points (minima, maxima, saddle points)

Gradient Properties: Key Insights

Definition: Machine Learning Connection

Optimization algorithms use gradients to:

- Find minimum loss (gradient descent: $\theta_{new} = \theta_{old} - \alpha \nabla L$)
- Navigate complex parameter spaces
- Escape saddle points
- Converge to optimal solutions

Summary: Contours and Gradients in ML

Key Points

What We Learned

- **Contour plots:** Visualize function behavior in 2D

Summary: Contours and Gradients in ML

Key Points

What We Learned

- **Contour plots:** Visualize function behavior in 2D
- **Different shapes:** Circular, diamond, hyperbolic, asymmetric

Summary: Contours and Gradients in ML

Key Points

What We Learned

- **Contour plots:** Visualize function behavior in 2D
- **Different shapes:** Circular, diamond, hyperbolic, asymmetric
- **Gradients:** Point toward steepest function increase

Summary: Contours and Gradients in ML

Key Points

What We Learned

- **Contour plots:** Visualize function behavior in 2D
- **Different shapes:** Circular, diamond, hyperbolic, asymmetric
- **Gradients:** Point toward steepest function increase
- **Perpendicular relationship:** Gradients contours

Summary: Contours and Gradients in ML

Important: ML Applications

- **Loss landscapes:** Understanding optimization challenges

Summary: Contours and Gradients in ML

Important: ML Applications

- **Loss landscapes:** Understanding optimization challenges
- **Gradient descent:** Following steepest descent direction

Summary: Contours and Gradients in ML

Important: ML Applications

- **Loss landscapes:** Understanding optimization challenges
- **Gradient descent:** Following steepest descent direction
- **Regularization:** L1/L2 penalties create different contour shapes

Summary: Contours and Gradients in ML

Important: ML Applications

- **Loss landscapes:** Understanding optimization challenges
- **Gradient descent:** Following steepest descent direction
- **Regularization:** L1/L2 penalties create different contour shapes
- **Saddle points:** Common in deep learning optimization

Summary: Contours and Gradients in ML

Important: ML Applications

- **Loss landscapes:** Understanding optimization challenges
- **Gradient descent:** Following steepest descent direction
- **Regularization:** L1/L2 penalties create different contour shapes
- **Saddle points:** Common in deep learning optimization

Definition: Next Steps

These concepts enable understanding of:

Summary: Contours and Gradients in ML

Important: ML Applications

- **Loss landscapes:** Understanding optimization challenges
- **Gradient descent:** Following steepest descent direction
- **Regularization:** L1/L2 penalties create different contour shapes
- **Saddle points:** Common in deep learning optimization

Definition: Next Steps

These concepts enable understanding of:

- Advanced optimization algorithms

Summary: Contours and Gradients in ML

Important: ML Applications

- **Loss landscapes:** Understanding optimization challenges
- **Gradient descent:** Following steepest descent direction
- **Regularization:** L1/L2 penalties create different contour shapes
- **Saddle points:** Common in deep learning optimization

Definition: Next Steps

These concepts enable understanding of:

- Advanced optimization algorithms
- Learning rate selection

Summary: Contours and Gradients in ML

Important: ML Applications

- **Loss landscapes:** Understanding optimization challenges
- **Gradient descent:** Following steepest descent direction
- **Regularization:** L1/L2 penalties create different contour shapes
- **Saddle points:** Common in deep learning optimization

Definition: Next Steps

These concepts enable understanding of:

- Advanced optimization algorithms
- Learning rate selection
- Convergence analysis